

DISEÑO DE LA ETAPA DE FRECUENCIA INTERMEDIA DE UN RADAR EN FPGA

Ing. David Álvarez Cañizares¹, DrCs Nelson Chávez Ferry²

¹Cujae, La Habana Cuba, ¹Cujae, La Habana Cuba
dvdalvarez1985@gmail.com

RESUMEN

El Grupo de Investigación de Radar del Departamento de Telecomunicaciones y Telemática de la Cujae posee un radar experimental instalado en el edificio número dos de la universidad que está conformado por partes de un SON-9A y un P-37, ambos de diseño y fabricación soviética. Estos dos radares son de generaciones atrasadas, pero de diseños excelentes en cuanto al sistema en sí. Sus partes y componentes son de tecnologías electrónicas, eléctricas y electromecánicas con una misión definida. Con la creciente necesidad de desarrollar nuevas plataformas de radar manteniendo relativamente bajos los costos, se necesita desarrollar un radar genérico compatible con múltiples plataformas de radar. Con el desarrollo de este trabajo se pretende dar una solución parcial a esta problemática a través del diseño de la etapa de FI de dicho radar utilizando un FPGA. En la misma se brindan detalles del diseño que se verifican utilizando la metodología Hardware In the Loop (HIL).

PALABRAS CLAVES: radar, software, FPGA, DDS, Diezmado, FIR, Filtro Digital (,).

DESIGN OF THE INTERMEDIATE FREQUENCY STAGE OF A RADAR IN FPGA

ABSTRACT

The Radar Research Group of the Department of Telecommunications and Telematics at Cujae has an experimental radar installed in the building number two of the university that is made up of parts of a SON-9A and P-37, both Soviet designed and made. These two radars are of backward generations, but of excellent designs as the system itself. Its parts and components are electronic, electrical and electromechanical technologies with a defined mission. With the growing need to develop new radar platforms while keeping costs relatively low, it is necessary to develop a compatible generic multi-platform radar. With the development of this work a partial solution to this problem is given by designing the IF stage of the radar using an FPGA. In this solution, details and verifications are made using the methodology Hardware in the loop (HIL).

KEY WORDS: radar, software, FPGA, DDS, NCO, Decimation, FIR, Digital Filter, Digital down conversion, representations, software.

INTRODUCCIÓN

Cualquier proyecto o tema de investigación en el campo de la Detección Remota por Ondas de Radio, fundamentalmente los de radar y los de radioastronomía, requieren de instalaciones de laboratorio actualizadas tecnológicamente que permitan realizar las observaciones y mediciones pertinentes. El Grupo de Investigación de Radar del Departamento de Telecomunicaciones y Telemática de la CUJAE posee un radar experimental (Radar Experimental de Laboratorio - REL) instalado en el edificio número dos de la universidad y conformado por partes de dos radares diferentes; un SON-9A y un P-37, ambos de diseño y fabricación soviética, que opera en la banda S (longitud de onda de 10 cm.). Estos dos radares son de generaciones atrasadas, pero de diseños excelentes en cuanto al sistema en sí. Sus partes y componentes son de tecnologías electrónicas, eléctricas y electromecánicas con una misión definida. Con la creciente necesidad de desarrollar nuevas plataformas de radar manteniendo relativamente bajos los costos, se necesita desarrollar un radar genérico compatible con múltiples plataformas de radar. Con este objetivo, se decidió en el grupo de investigación la sustitución de algunas partes del radar por etapas digitales que permitan el uso de tecnologías ya establecidas en el mercado como las de los SDR y los FPGA.

Una de las partes que requiere una modernización a través de la digitalización es la etapa amplificadora de frecuencia intermedia (FI). Esta etapa del receptor del radar, al estar diseñada como filtro cuasi adaptado, es la que garantiza la máxima relación señal a ruido a su salida en FI. Digitalizar esta etapa supone la sustitución del ruido interno del receptor por ruido de cuantificación, lo que constituye un paso necesario para obtener una probabilidad de detección máxima. La digitalización de esta etapa del receptor permitirá también disminuir tanto el tiempo como el costo de desarrollo en el grupo de investigación pues brinda la posibilidad de realizar el resto de las etapas del receptor en un sistema digital lográndose con ello la reutilización del hardware. Además, de esta forma se puede utilizar el mismo diseño para un amplio espectro de frecuencias intermedias de radar y anchos de banda permitiendo con ello la sustitución de la etapa de RF solamente acomodando algunos de sus parámetros. De esta forma la etapa de FI que se pretende sustituir tendría la función de interfaz entre el procesamiento de RF analógico y el procesamiento digital en banda base y de esto se desprende la gran importancia de la misma.

El filtro adaptado se puede realizar de dos formas fundamentales. La primera es implementar un filtro adaptado a la señal pasa banda y la segunda es implementar un filtro adaptado a la señal en banda base. En tiempo discreto, el filtro adaptado se puede implementar con un filtro FIR cuyos coeficientes son las muestras de la respuesta al impulso del filtro adaptado en tiempo continuo. De esta forma, el número de coeficientes de este FIR está en dependencia de la duración máxima de la señal de sondeo y la frecuencia de muestreo del ADC. A su vez, la frecuencia de muestreo tiene que ser mayor que el doble de la máxima frecuencia de la señal. La señal en FI se encuentra centrada en 30 MHz y tiene un ancho de banda máximo de 2 MHz. Con lo anterior se puede notar que la frecuencia de muestreo necesaria para digitalizar la señal en FI debe ser en teoría mayor que 62 Msps. Sin embargo, la señal de sondeo tiene en realidad un ancho de banda mucho menor (1 MHz) en banda base. Si se aprovecha lo anterior, se puede realizar una reducción de la frecuencia de muestreo si se lleva la señal a banda base necesitándose menor cantidad de coeficientes para el cálculo del filtro adaptado a la señal de sondeo. Además, tener una frecuencia de reloj del sistema mucho mayor que la frecuencia de muestreo de la señal, permite

reducir el consumo de recursos en el FPGA reutilizando algunos componentes en diferentes ciclos de reloj.

De esta manera, el esquema básico del sistema que se pretende implementar es el mostrado en la figura 1, reutilizando la etapa de RF existente. Así, la tarea a cumplir estará fundamentalmente orientada a la etapa digital donde se realizará la mayor parte del procesamiento de FI.

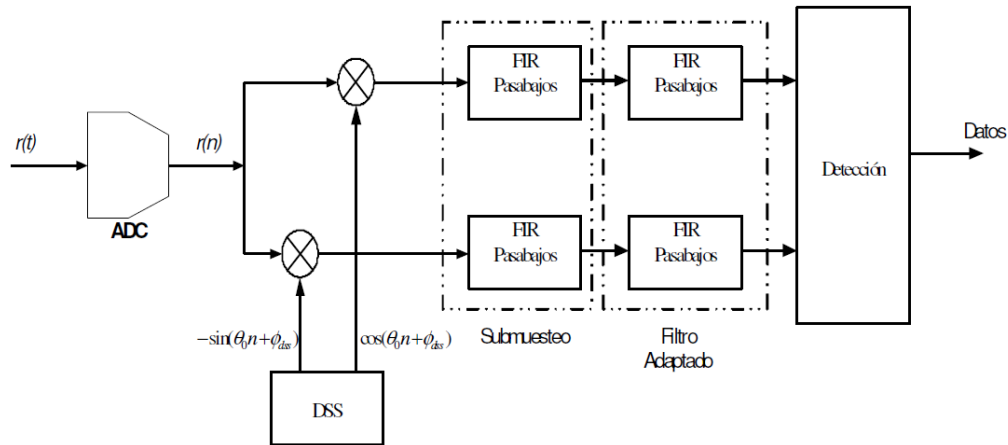


Figura 1: Esquema Básico del Demodulador. Se incluyen además el filtro adaptado a la componente en fase y en cuadratura y la etapa de decisión.

DISEÑO DE LA ETAPA DE FI

Diseño de un DDS en Simulink para su implementación en un FPGA

La síntesis digital directa de frecuencias, por definición, consiste en generar una representación digital de la señal deseada, a partir de una única referencia de frecuencia fija. La técnica más frecuentemente utilizada para su implementación es el empleo de una tabla de búsqueda directa, consistente en almacenar en una memoria las muestras de amplitud de una senoide para incrementos de fase sucesivos [1]. De esta forma la frecuencia f_0 de la señal a la salida controlada por el incremento de fase M y la frecuencia del reloj f_{clk} se calcula como se muestra en la ecuación 1:

$$f_0 = \frac{M \cdot f_{clk}}{2^N} \quad (1)$$

El diseño del DDS se realizó en el Simulink de Matlab utilizando los componentes del System Generator de Xilinx. Para ello sólo se necesitan un sumador y un retardo que se utilizan como acumulador de fase, un bloque Slice para tomar los bits más significativos de la salida del acumulador y dos segmentos de memoria ROM o RAM según se desee [1]. Todo lo anterior se puede observar en la figura 2.

DISEÑO DE LA ETAPA DE FRECUENCIA INTERMEDIA DE UN RADAR EN FPGA

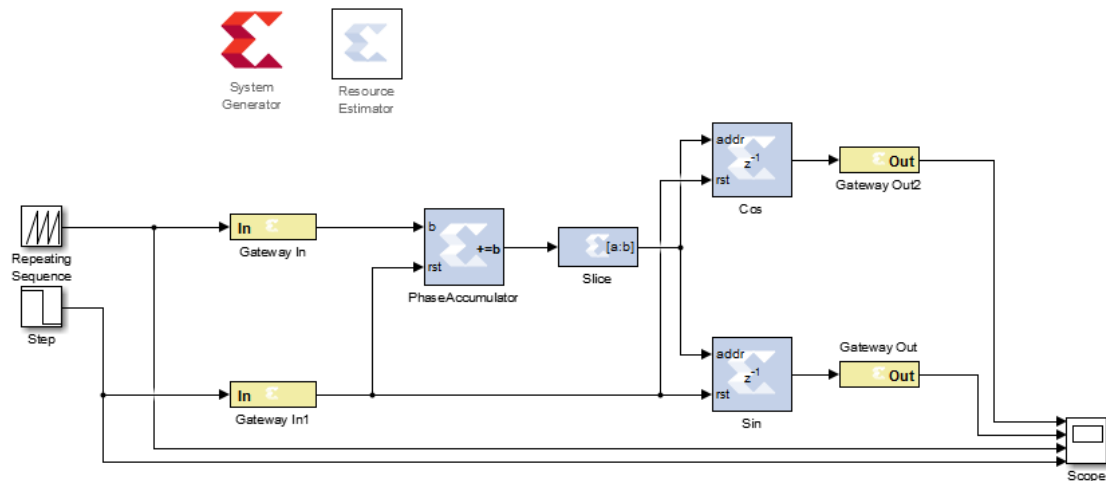


Figura 2: Implementación en el System Generator de Xilinx del DDS.

Para comprobar que el diseño es válido, se realizó una simulación utilizando una señal de habilitación y una señal rampa para aumentar el valor en la entrada del acumulador de fase. Los resultados de dicha simulación pueden observarse en la figura3.

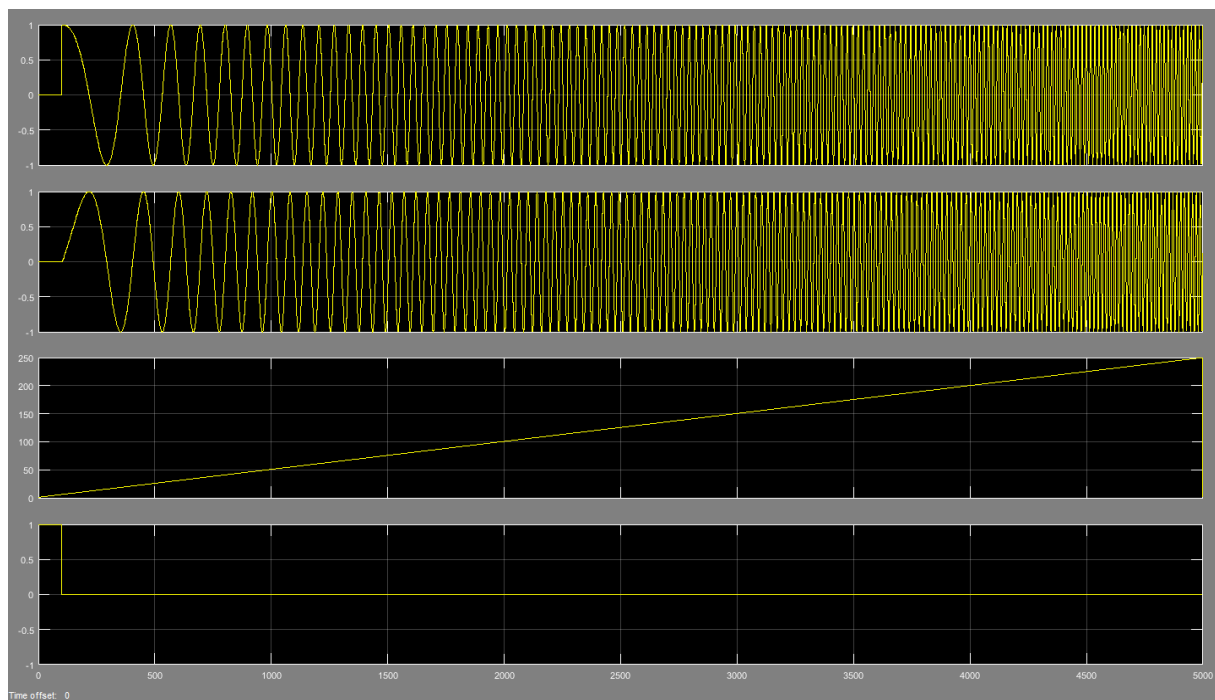


Figura 3: Simulación para la verificación del funcionamiento del DDS.

Como puede observarse en la figura, aumentar el valor de entrada al acumulador de fase provoca un aumento de la frecuencia a la salida del DDS. Nótese también la diferencia de fase en el coseno y el seno, así como el funcionamiento de la habilitación del sistema. Esta prueba se realizó también analizando el espectro de la señal observándose la variación de una señal impulsiva en el tiempo.

Diseño de un filtro pasa bajo de diezmado en Simulink para su implementación en un FPGA

Para obtener la señal en banda base, se necesita filtrar pasa bajas y diezmar la frecuencia de muestreo. Esto se debe a que la frecuencia de muestreo es de 100 Msps y el ancho de banda de la señal que se desea procesar es siempre menor 2 MHz centrada en 30 MHz. Con una frecuencia de muestreo de 100 Msps el límite de Nyquist sería 50 MHz. Con el ancho de banda y la frecuencia a la que se encuentra centrada la señal, se puede calcular la máxima frecuencia de interés en la etapa de FI la cual es de 31 MHz. Esta frecuencia máxima de interés es posible muestrearla pues el límite impuesto por esta frecuencia de muestreo es aproximadamente 1.5 veces mayor.

Un tipo de filtro pasa bajas de diezmado utilizado comúnmente son los filtros FIR (Finite Impulse Response por sus siglas en inglés). Este filtro elimina todas las componentes de la señal que se encuentran más allá de su banda de paso manteniendo una respuesta de fase lineal [2]. Diezmar por un factor M la frecuencia de muestreo, no es más que retener la M – ésima muestra de la entrada en la salida [2]. La salida de un FIR viene dada por la ecuación 2 según [2].

$$y[n] = \sum_{i=0}^N h_i \cdot x[n - i] \quad (2)$$

donde h_i son los coeficientes del filtro, N es la cantidad de coeficientes y $x[n]$ son las muestras de la señal a la entrada del filtro.

Para el diseño de esta etapa se tomó la decisión de disminuir la frecuencia de muestreo por un factor entero de 25 obteniéndose una frecuencia de muestreo de 4 Msps. Dicha operación se puede realizar en una o varias etapas.

El primer paso a seguir es el diseño del filtro FIR pasa bajo que se utilizará para limitar la señal que se quiere diezmar. Para ello se, se diseñó un filtro de ventana de Kaiser con 25 coeficientes cuya frecuencia de corte se encuentra en $\pi/5$ en la figura 4. La forma en que se realiza este diseño puede encontrarse bien explicada en [2].

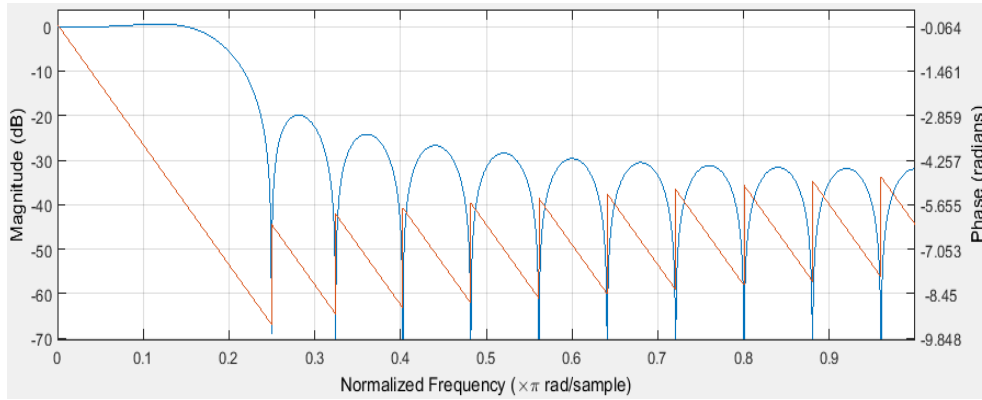


Figura 4: Respuesta de amplitud y fase de un FIR de orden 24 diseñado mediante el método de ventana de Kaiser.

Una vez diseñado el filtro a utilizar, se debe pasar a descomponer el filtro en cinco filtros polifásicos. Para descomponer el filtro se deben tomar muestras diezmadas de la respuesta al impulso del filtro diseñado como se muestra en la figura 5. La salida de los filtros polifásicos aplicados a la descomposición polifásica de la señal de entrada está definida por las ecuaciones 3 según se explica en [3] y [4].

$$\begin{aligned}
 y_{M-1}[n] &= h_0[n] \otimes x_{M-1}[n] \\
 y_{M-2}[n] &= h_1[n] \otimes x_{M-2}[n] \\
 &\vdots \\
 y_0[n] &= h_{M-1}[n] \otimes x_0[n] \\
 y[n] &= y_0[n] + y_1[n] + \dots + y_{M-1}[n]
 \end{aligned} \tag{3}$$

donde \otimes denota la operación de convolución (filtrado) y la señal de entrada y el filtro se descomponen según las ecuaciones 4. Proceso también explicado en [3] y [4].

$$\begin{aligned}
 h_p[n] &= h[p + Mn] \\
 x_p[n] &= x[p + Mn]
 \end{aligned} \tag{4}$$

donde $p=0, 1, \dots, M-1$

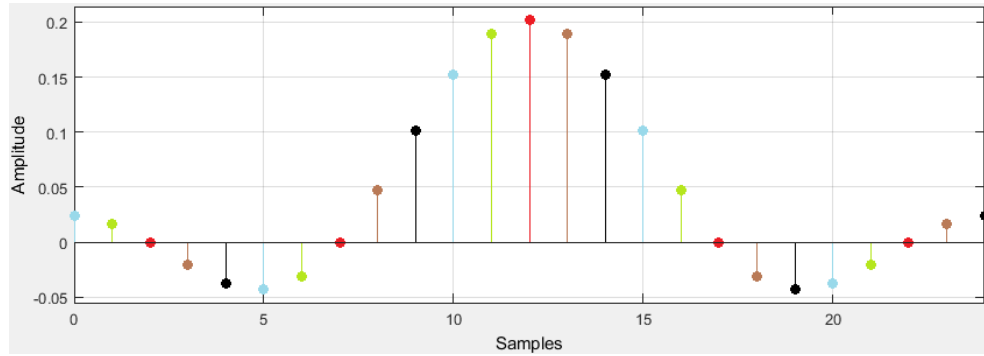


Figura 5: Respuesta al impulso del filtro FIR diseñado. Se representa con diferentes colores la descomposición polifásica de este filtro

Aprovechando la simetría de los filtros polifásicos y la diferencia de frecuencia de muestreo entre el reloj y la señal a la entrada, se pueden calcular las respuestas de los FIR utilizando el sistema que se muestra en la figura 6

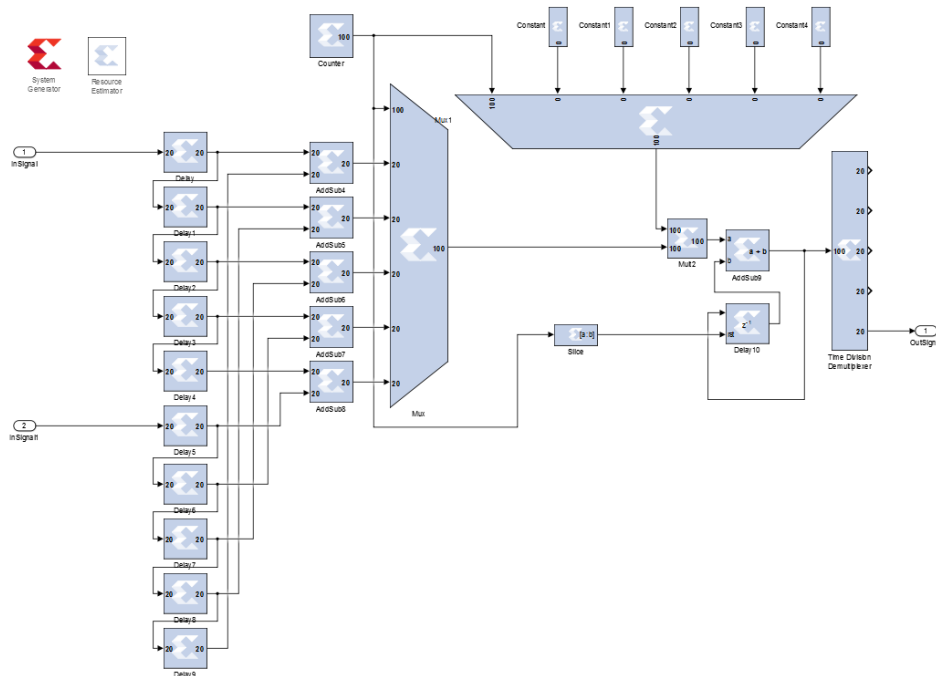


Figura 6: Diseño del diezmador para optimizar el consumo de multiplicadores y sumadores realizando las operaciones secuencialmente mediante el uso de multiplexores.

El cambio fundamental si se compara con un FIR típico se encuentra en la sustitución de la etapa de multiplicadores por un solo multiplicador que realiza todas las operaciones secuencialmente de forma similar a como se explica en [5] y [6]. Esto se logra mediante el uso de multiplexores tanto para los coeficientes del filtro (pueden almacenarse los coeficientes en ROM o RAM) como para las muestras a la entrada. En la medida en que se van realizando las multiplicaciones se deben ir sumando. Esto se puede lograr sumando la multiplicación en curso con la anterior utilizando un sumador y un retardador. Para el correcto funcionamiento del sistema se debe sumar con cero la primera operación. Para ello se necesita resetear el retardador cuando se realiza la última suma. Esto se logra utilizando el desborde del contador que maneja los multiplexores. En la figura 7, se muestra la comparación entre la variante propuesta y el FIR que se brinda en el System Generator de Xilinx con un diezmador en la salida.

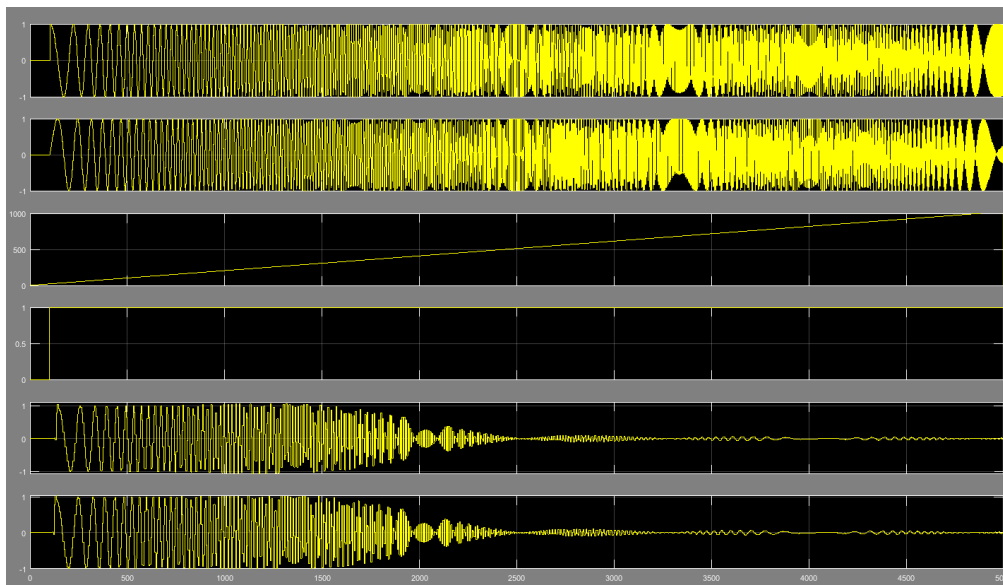


Figura 7: Señales de salida del sistema para realizar las pruebas. De arriba hacia abajo: señal coseno, señal seno, señal de variación de la fase, señal de habilitación, señal a la salida de un FIR típico con un diezmador y señal de salida del diezmador que realiza las operaciones secuencialmente.

Con la similitud entre las dos salidas mostradas en la figura, se puede validar el diseño a nivel de simulación. Con esta variante se logra minimizar el consumo de recursos utilizándose muchos menos multiplicadores.

Diseño de un filtro adaptado en Simulink para su implementación en un FPGA

Un filtro adaptado es un sistema lineal invariante cuya función principal es detectar la presencia de una señal conocida, o de referencia, dentro de una señal recibida [7].

Una vez fijadas las muestras de la señal a la cual se debe adaptar el filtro en banda base, el filtro adaptado se obtiene filtrando la señal recibida con un FIR cuya respuesta al impulso es la conjugada de

la señal conocida invertida en el tiempo [7] como se muestra en la ecuación 5. Con el objetivo de que el diseño final permita cambiar la señal de sondeo, se desea además que los coeficientes del filtro sean programables.

$$h(t) = Ks^*(t_0 - t) \quad (5)$$

Existe un criterio de diseño para el cual la duración del pulso de radiofrecuencia es inversamente proporcional al ancho de banda de la señal [8]. Para el caso del P-37, el tiempo de duración del pulso puede ser de $2.5 \mu s$ a $2.9 \mu s$ o de $1.7 \mu s \pm 0.3 \mu s$ en dependencia del régimen de trabajo [9].

Con el mayor tiempo de duración del pulso, se puede calcular el número de coeficientes necesarios para implementar un filtro adaptado a la señal de sondeo, multiplicándolo por la frecuencia de muestreo. De esta manera, utilizando un pulso de $3.0 \mu s$, se necesitan 300 coeficientes para una frecuencia de muestreo de 100 Msps mientras que para una frecuencia de muestreo de 4 Msps se necesitan solo 12 coeficientes.

Si se aprovecha la disminución de la frecuencia de muestreo de la señal, teniendo en cuenta que la frecuencia del reloj continúa siendo la misma, se pueden realizar las operaciones de multiplicación para el cálculo de la respuesta del FIR en varios ciclos de forma similar a como se realiza el filtro pasa bajas de diezmado visto con anterioridad. De esta forma, disminuyendo la frecuencia de muestreo de 100 Msps a 4 Msps, es posible realizar 25 multiplicaciones en un período de muestreo. Lo anterior permite que con un solo multiplicador se pueda efectuar el filtrado con un FIR de orden 25. Estos 25 coeficientes son equivalentes a un tiempo de duración del pulso de $6.25 \mu s$ que es mucho mayor que el máximo requerido para el P-37.

El filtro adaptado a la señal en banda base, es un filtro adaptado a una señal compleja [7, 10]. De esta manera, se necesita un filtro adaptado a la señal I y un filtro adaptado a la señal Q (figura 8).

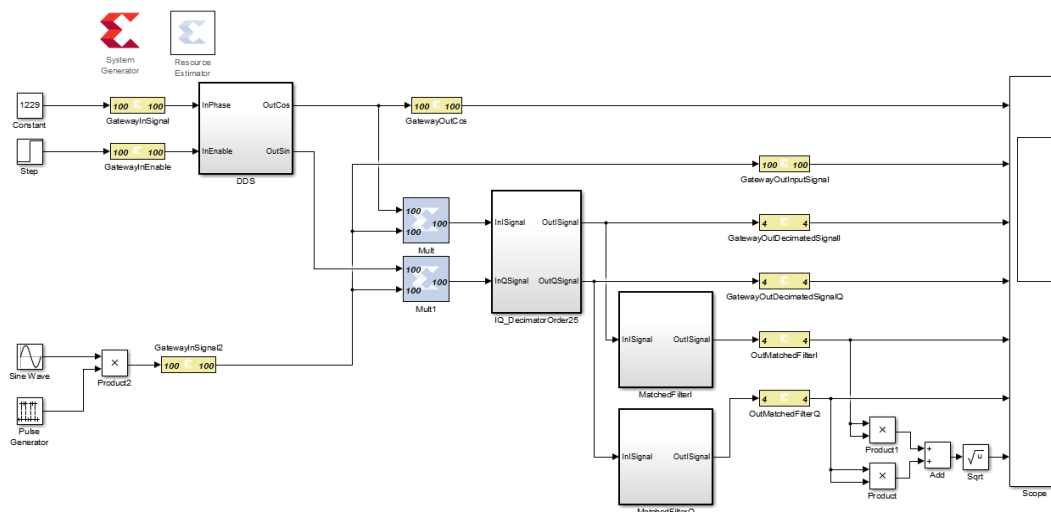


Figura 8: Sistema diseñado para sustituir la etapa de FI del REL. Se incluye un DDS, una etapa de diezmado y un filtro adaptado a la señal compleja en banda base

Para el caso de un pulso de radio frecuencia, la señal en banda base es un pulso rectangular [7]. En general, en dependencia de la diferencia de fase entre la señal recibida y la generada en el DDS, la amplitud del pulso variará en las salidas I y Q. Por esta razón, si el filtro se encuentra adaptado a una señal con una fase determinada, no detectará la presencia de un pulso con otra fase (o será poco detectable). Para solucionar esta problemática, se puede utilizar un filtro adaptado cuya respuesta al impulso sea un pulso rectangular tanto en la parte real como en la imaginaria. Luego se calcula el valor de la amplitud obteniéndose un filtro adaptado que no tiene en cuenta la fase de la señal de entrada como puede observarse en la respuesta de dicho sistema mostrada en la figura 9.

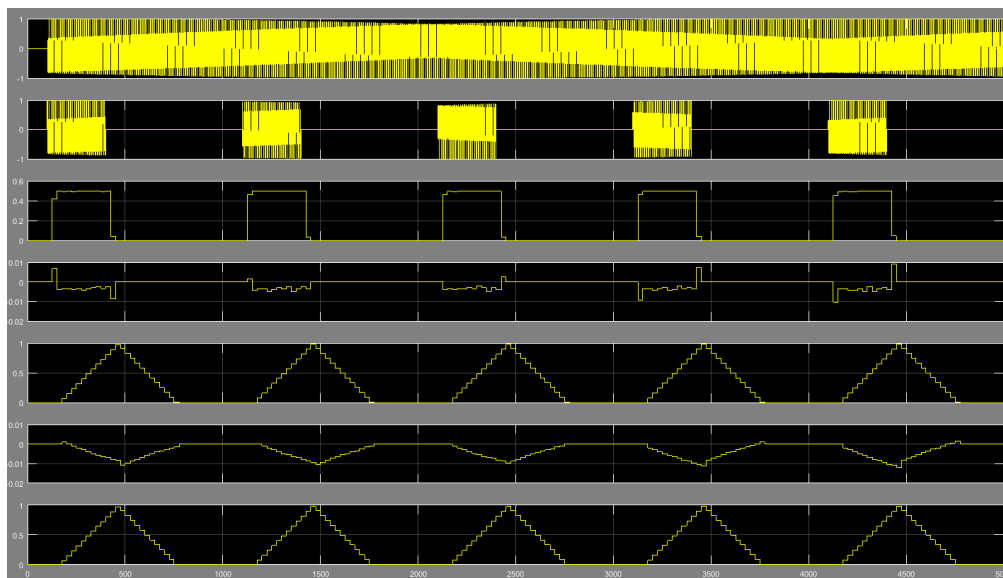


Figura 9: Señales de entrada y salida del sistema diseñado para sustituir la etapa de FI del REL. De arriba hacia abajo: Señal coseno generada con el DDS, tren de pulsos de RF generados en Matlab, señal de I, señal Q, señal a la salida del filtro adaptado a I, señal a la salida del filtro adaptado a Q y la señal de amplitud de la salida del filtro adaptado.

En la figura se destaca la respuesta del filtro adaptado que es muy similar a la que se obtendría sin desplazamiento Doppler si se compara con la respuesta del filtro adaptado a un pulso de RF teórica mostrada en [7].

RESULTADOS DE IMPLEMENTACIÓN Y SIMULACIÓN

Co – Simulación en Hardware y Matlab

El proceso de Co – Simulación se inicia con la construcción del prototipo desarrollado completamente a base de bloques en System Generator. Todas las señales que serán inyectadas como estímulos para la validación del prototipo, deben ser integradas al diseño por medio de bloques Gateway In/Out, los cuales funcionan como fronteras de diseño. Estableciendo los bloques que serán implementados dentro del FPGA.

Para realizar las validaciones de esta forma, se genera la señal de excitación y se procesa la señal compleja IQ en el Workspace de Matlab. La señal se crea con un código realizado en Matlab que crea un tren de pulso en el cual se varía la frecuencia de cada pulso consecutivo (figura 10).

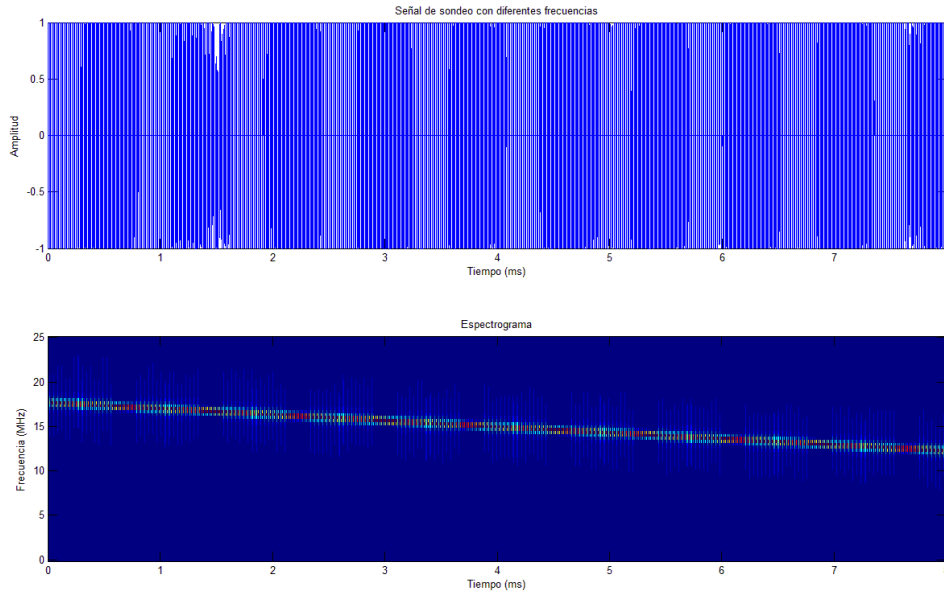


Figura 10: Señal de excitación para probar el comportamiento del sistema cuando se excita con una señal fuera de la banda de sintonización.

Para esta prueba se obtiene una gráfica de la función de ambigüedad y se puede estudiar cómo se comporta el sistema ante estímulos fuera de la banda de trabajo. La respuesta del sistema a la misma puede apreciarse en la figura 11.

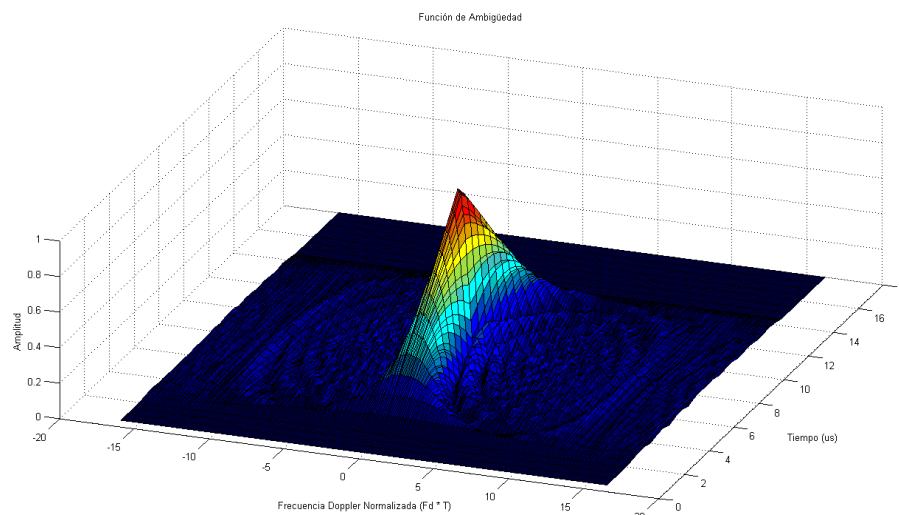


Figura 11: Función de ambigüedad normalizada obtenida utilizando Co - Simulación con el FPGA.

Si observa una vista superior de la gráfica de la función de ambigüedad normalizada obtenida como se muestra en la figura 12, se puede verificar como el producto $Tf_D \approx 1$ para el primer nulo Doppler. Estos resultados verifican la validez del diseño si se comparan una vez más con la respuesta del filtro adaptado a un pulso de RF teórica mostrada en [7].

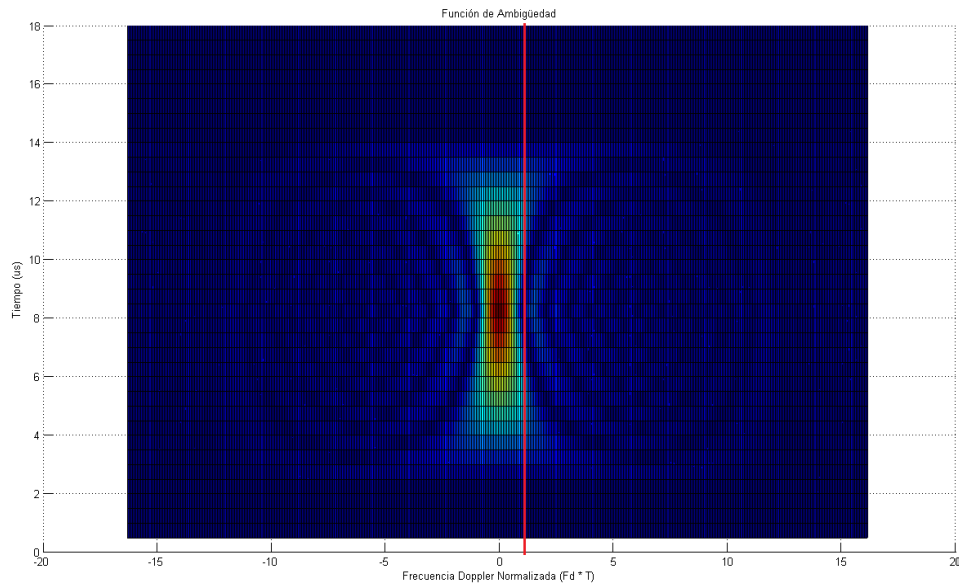


Figura 12: Vista superior de la función de ambigüedad normalizada.

Es válido aclarar que esta prueba se realiza sólo para verificar que el diseño cumple con la salida esperada teórica. No se debe pensar que la señal puede tener un corrimiento Doppler de esta magnitud pues el mismo es excesivo.

Consumo de Hardware

En la tabla 1 se muestra un resumen del consumo de hardware para la tarjeta de desarrollo utilizada. Esta información se obtiene del fichero xflow.log luego de generar el bloque de Co - simulación.

Tabla 1: Rango de valores de velocidades

| | | |
|-------------------------|--------------|-----|
| Número de BUFGMUXs | 4 de 24 | 16% |
| Número de MULT18X18SIOs | 6 de 20 | 30% |
| Número de RAMB16s | 7 de 20 | 35% |
| Número de Slices | 1085 de 4656 | 23% |
| Número de SLICEMs | 48 de 2328 | 2% |

De este resumen se destaca que el consumo de recursos nunca sobrepasa del 35 % lo que permite incorporar más etapas de procesamiento en el FPGA. Como puede observarse, una de las principales

fortalezas de este diseño es el bajo consumo de multiplicadores (solo 6 de los cuales 2 se utilizan en la traslación a banda base de la señal de entrada) teniendo en cuenta que se utilizan cuatro FIR de orden 25 para lo cual se requerirían aproximadamente 100 multiplicadores.

CONCLUSIONES.

Durante la etapa de diseño se concluyó que era mejor utilizar un demodulador IQ y un filtro adaptado a la señal en banda base en lugar de utilizar un filtro adaptado a la señal pasa banda directamente. A esta conclusión se llegó buscando minimizar el consumo de recursos en el FPGA. De las pruebas realizadas al filtro pasa bajo de diezmado se verificó que se debía utilizar una etapa de diezmado en lugar de dos para minimizar el ruido introducido por las operaciones con punto fijo en los filtros de respuesta finita al impulso. Al término de este trabajo se fijó el orden de dicho filtro utilizándose 25 coeficientes de forma tal que se necesitara un solo multiplicador en cada filtro. De igual forma, se diseñó el filtro adaptado a la señal en banda base. Para este se escogieron también 25 coeficientes con lo cual se satisface el tiempo de duración del pulso de RF y solo se necesita un multiplicador para computar su respuesta en un instante dado.

Por último, se validó el diseño utilizando la técnica denominada Hardware In the Loop, estimulando el sistema diseñado embebido en el FPGA con una señal construida en Matlab y comparando su respuesta con la teoría mostrada en [7].

REFERENCIAS.

1. Generador de señales con forma de onda arbitraria y ruido usando DDS en FPGA.
2. Oppenheim, A. V., Schafer, R. W. and Buck, J.R., Discrete - Time Signal Processing. 1999
3. Porteous M., Introduction to Digital Resampling, 2007
4. Proakis, J. G. and Manolakis, D. G., Digital Signal Processing, Principles Algorithms and Applications. 4th Edition, Prentice Hall, 2007.
5. Martínez Castro, L., Diseño de filtro digital múltiple tipo FIR utilizando FPL para su uso en un SDR. 2010.
6. Toledo de la Garza, K., Implementación en VHDL de Detector de Envolvente para BFSK. 2012.
7. Levannon, N. and E. Mozeson, Radar signals. 2004.
8. Brimeyer, T., et al. A generic radar processor design using software defined radar. 2007.
9. Radar de banda centimétrica 1ra parte, ITM
10. Stewart, B., et al., Software defined radio using Matlab & Simulink and the rtl - sdr. 2015, Strathclyde Academic Media.